

# Stochastic Programs with First-Order Dominance Constraints Induced by Mixed-Integer Linear Recourse

Ralf Gollmer, Frederike Neise, Rüdiger Schultz\*  
 Department of Mathematics  
 University of Duisburg-Essen, Campus Duisburg  
 Lotharstr. 65, D-47048 Duisburg, Germany

## Abstract

We propose a new class of stochastic integer programs whose special features are dominance constraints induced by mixed-integer linear recourse. For these models, we establish closedness of the constraint set mapping with the underlying probability measure as parameter. In the case of finite probability spaces, the models are shown to be equivalent to large-scale, block-structured, mixed-integer linear programs. We propose a decomposition algorithm for the latter and discuss preliminary computational results.

**Key Words.** Stochastic integer programming, stochastic dominance, mixed-integer optimization.

**AMS subject classifications.** 90C15, 90C11, 60E15.

## 1 Introduction

Stochastic programming models are derived from random optimization problems with information constraints. In the present paper we start out from the following random mixed-integer linear program

$$\min\{c^\top x + q^\top y : Tx + Wy = z(\omega), x \in X, y \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}\}, \quad (1)$$

together with the information constraint that  $x$  must be selected without anticipation of  $z(\omega)$ . This leads to a two-stage scheme of alternating decision and observation: The decision on  $x$  is followed by observing  $z(\omega)$  and then  $y$  is taken, thus depending on  $x$  and  $z(\omega)$ . Accordingly,  $x$  and  $y$  are called first- and second-stage decisions, respectively.

Assume that the ingredients of (1) have conformable dimensions, that  $W$  is a rational matrix, and that  $X \subseteq \mathbb{R}^m$  is a nonempty polyhedron, possibly involving integer requirements to components of  $x$ .

The mentioned two-stage dynamics becomes explicit by the following reformulation of (1)

$$\begin{aligned} \min_x \left\{ c^\top x + \min_y \{ q^\top y : Wy = z(\omega) - Tx, y \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'} \} : x \in X \right\} \\ = \min_x \{ c^\top x + \Phi(z(\omega) - Tx) : x \in X \} \end{aligned} \quad (2)$$

where

$$\Phi(t) := \min\{q^\top y : Wy = t, y \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}\}. \quad (3)$$

The function  $\Phi$ , called the value function of the mixed-integer linear program

$$\min\{q^\top y : Wy = t, y \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}\},$$

has been studied in parametric optimization. Under the assumptions

$$(A1) \quad (\text{complete recourse}) \quad W(\mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}) = \mathbb{R}^s,$$

---

\*schultz@math.uni-duisburg.de

(A2) (sufficiently expensive recourse)  $\{u \in \mathbb{R}^s : W^\top u \leq q\} \neq \emptyset$ ,

it holds that  $\Phi$  is real-valued and lower semicontinuous on  $\mathbb{R}^s$ , i.e.,  $\liminf_{t_n \rightarrow t} \Phi(t_n) \geq \Phi(t)$  for all  $t \in \mathbb{R}^s$ , [2, 6].

In view of (2), the random optimization problem (1) gives rise to the family of random variables

$$\left( c^\top x + \Phi(z(\omega) - Tx) \right)_{x \in X}. \quad (4)$$

Thus every first-stage decision  $x \in X$  induces a random variable  $f(x, \omega) := c^\top x + \Phi(z(\omega) - Tx)$ . Traditional two-stage stochastic programming aims at optimizing nonanticipative decisions, i.e., finding a “best”  $x$ , or in other words a “best” member in the family (4) of random variables. For the specification of “best”, statistical parameters reflecting mean and/or risk are employed. Early approaches in the literature used the expectation, leading to optimization problems

$$\min\{\mathbb{E}[f(x, \omega)] : x \in X\}. \quad (5)$$

Employing the weighted sum of  $\mathbb{E}$  and some risk measure  $\mathcal{R}$  leads to mean-risk models

$$\min\{\mathbb{E}[f(x, \omega)] + \rho \cdot \mathcal{R}[f(x, \omega)] : x \in X\} \quad (\rho > 0 \text{ fixed}). \quad (6)$$

There is an extensive literature on structural analysis and algorithm design for this class of stochastic programs, see for instance [1, 5, 12, 16, 17, 19, 23, 25, 27, 28].

Here, we take an alternative view. Rather than heading for “best” members of (4), we want to identify “acceptable” members, and optimize over them. This leads to a new class of stochastic integer programs, see (8) below, whose structural analysis and algorithmic treatment is the aim of the present paper.

Stochastic dominance, an established concept in decision theory ([13, 20, 21]), provides a possibility to formalize the above mentioned “acceptability”. In the present paper we deal with first-order stochastic dominance. When preferring small outcomes to big ones, a (real-valued) random variable  $X$  is said to dominate a random variable  $Y$  to first order ( $X \succeq_1 Y$ ) iff  $\mathbb{E}h(X) \leq \mathbb{E}h(Y)$  for all nondecreasing functions  $h$  for which both expectations exist. An equivalent formulation reads as follows (see, e.g., [21]):

$$X \succeq_1 Y \quad \text{iff} \quad \mathbb{P}[\{\omega : X(\omega) \leq \eta\}] \geq \mathbb{P}[\{\omega : Y(\omega) \leq \eta\}] \quad \forall \eta \in \mathbb{R}. \quad (7)$$

Coming back to our two-stage random optimization problem (1) and the related family (4), we assume some (random) benchmark cost profile  $d(\omega)$  be given. We consider only those  $x \in X$  “acceptable” for which the corresponding  $f(x, \omega)$  dominates to first order the benchmark profile  $d(\omega)$ . Over all “acceptable”  $x \in X$  we optimize some function  $g : \mathbb{R}^m \rightarrow \mathbb{R}$ . This leads to the following stochastic program with first-order dominance constraint induced by mixed-integer linear recourse

$$\min\{g(x) : f(x, \omega) \succeq_1 d(\omega), x \in X\}. \quad (8)$$

Stochastic optimization problems with dominance constraints involving general random variables were pioneered in [9, 10, 11, 22]. These papers address structure, stability, and algorithms for (8) if general random variables, enjoying suitable continuity, smoothness, or linearity properties in  $x$  and/or  $\omega$ , are placed instead of  $f(x, \omega)$ . The random variables  $f(x, \omega)$ , on the one hand, are more specific, since essentially given by the mixed-integer value function in (3). On the other hand, the results from [9, 10, 11, 22] are not applicable to our setting due to lacking smoothness of  $\Phi$ , recall the above mentioned lower semicontinuity.

Our paper is organized as follows. In Section 2 we elaborate a basic structural property of the constraint sets of (8) and draw some conclusions, among others, on the stability behaviour of (8). Section 3 is devoted to algorithmic aspects. We prove equivalence of (8) with a structured mixed-integer linear program if the underlying probability spaces are finite. Then we propose a decomposition algorithm for these models. The paper concludes with Section 4 where we report some first computational results.

## 2 Structure and Stability

The stochastic program (8) is essentially governed by its constraint set. In what follows we establish some results concerning the basic well-posedness of this set.

Let  $\mathcal{P}(\mathbb{R}^s), \mathcal{P}(\mathbb{R})$  be the sets of all Borel probability measures on  $\mathbb{R}^s$  and  $\mathbb{R}$ , respectively. By  $\mu \in \mathcal{P}(\mathbb{R}^s)$  and  $\nu \in \mathcal{P}(\mathbb{R})$  we denote the probability measures induced by the random variables  $z(\omega)$  and  $d(\omega)$ . We fix  $\nu$  and consider the multifunction  $C : \mathcal{P}(\mathbb{R}^s) \rightarrow 2^{\mathbb{R}^m}$  where

$$C(\mu) := \{x \in \mathbb{R}^m : f(x, z) \succeq_1 d, x \in X\}.$$

Moreover, we equip  $\mathcal{P}(\mathbb{R}^s)$  with weak convergence of probability measures ([3]). A sequence  $\{\mu_n\}$  in  $\mathcal{P}(\mathbb{R}^s)$  is said to converge weakly to  $\mu \in \mathcal{P}(\mathbb{R}^s)$ , written  $\mu_n \xrightarrow{w} \mu$ , if for any bounded continuous function  $h : \mathbb{R}^s \rightarrow \mathbb{R}$  it holds  $\int_{\mathbb{R}^s} h(z) \mu_n(dz) \rightarrow \int_{\mathbb{R}^s} h(z) \mu(dz)$  as  $n \rightarrow \infty$ .

**Proposition 2.1** *Assume (A1) and (A2). Then  $C$  is a closed multifunction on  $\mathcal{P}(\mathbb{R}^s)$ . This means that for arbitrary  $\mu \in \mathcal{P}(\mathbb{R}^s)$  and sequences  $\mu_n \in \mathcal{P}(\mathbb{R}^s), x_n \in C(\mu_n)$  with  $\mu_n \xrightarrow{w} \mu$  and  $x_n \rightarrow x$  it follows that  $x \in C(\mu)$ .*

**Proof:** In view of  $x_n \in C(\mu_n)$  and (7) it holds for all  $n$

$$\nu[d \leq \eta] \leq \mu_n[f(x_n, z) \leq \eta] \quad \forall \eta \in \mathbb{R}. \quad (9)$$

(The shorthand notations  $d \leq \eta$  and  $f(x_n, z) \leq \eta$  refer to the sets  $\{d \in \mathbb{R} : d \leq \eta\}$  and  $\{z \in \mathbb{R}^s : f(x_n, z) \leq \eta\}$ , respectively.)

Denote  $M_\eta(x) := \{z \in \mathbb{R}^s : f(x, z) > \eta\}$ . By (A1) and (A2), the function  $\Phi$ , and hence  $f(x, \cdot)$ , is lower semicontinuous. Therefore,  $M_\eta(x)$  is open for all  $\eta \in \mathbb{R}$  and all  $x \in \mathbb{R}^m$ . With the new notation, (9) says that for all  $n$

$$\nu[d \leq \eta] + \mu_n[M_\eta(x_n)] \leq 1 \quad \forall \eta \in \mathbb{R}. \quad (10)$$

Since  $M_\eta(x)$  is open, the Portmanteau Theorem (see [3], Theorem 2.1, pp. 11/12) implies that

$$\mu[M_\eta(x)] \leq \liminf_n \mu_n[M_\eta(x_n)] \quad \forall \eta \in \mathbb{R}. \quad (11)$$

The lower semicontinuity of  $\Phi$  yields

$$M_\eta(x) \subseteq \liminf_n M_\eta(x_n) \quad \forall \eta \in \mathbb{R}. \quad (12)$$

Here “ $\liminf_n$ ” denotes the set-theoretic limes inferior, i.e., the set of all points belonging to all but a finite number of sets  $M_\eta(x_n)$ . For fixed  $n$ , (12) and the lower semicontinuity of the probability measure (see [4], Theorem 4.1, p. 48) now imply

$$\mu_n[M_\eta(x)] \leq \mu_n[\liminf_k M_\eta(x_k)] \leq \liminf_k \mu_n[M_\eta(x_k)] \quad \forall \eta \in \mathbb{R}.$$

Taking the limes inferior with respect to  $n$ , we obtain

$$\liminf_n \mu_n[M_\eta(x)] \leq \liminf_n \liminf_k \mu_n[M_\eta(x_k)] \leq \liminf_n \mu_n[M_\eta(x_n)] \quad \forall \eta \in \mathbb{R}. \quad (13)$$

For the last inequality we have picked the diagonal subsequence where  $n = k$ . Putting together (11) and (13) we arrive at

$$\mu[M_\eta(x)] \leq \liminf_n \mu_n[M_\eta(x_n)] \quad \forall \eta \in \mathbb{R}. \quad (14)$$

Taking the limes inferior with respect to  $n$  in (10) and observing (14) leads to

$$\nu[d \leq \eta] + \mu[M_\eta(x)] \leq \nu[d \leq \eta] + \liminf_n \mu_n[M_\eta(x_n)] \leq 1 \quad \forall \eta \in \mathbb{R}.$$

This implies, see (10), (9), and (7), that  $f(x, z) \succeq_1 d$ . By the closedness of  $X$ ,  $x_n \rightarrow x$ , and  $x_n \in X$  (for all  $n$ ) we have that  $x \in X$ . Altogether it follows that  $x \in C(\mu)$ , and the proof is complete.  $\square$

**Remark 2.2** (About variable  $\nu$ .) *Equipping  $\mathcal{P}(\mathbb{R})$  with uniform convergence of distribution functions (Kolmogorov-Smirnov convergence), as for instance in [9], allows to extend Proposition 2.1 to the multifunction  $\bar{C} : \mathcal{P}(\mathbb{R}^s) \times \mathcal{P}(\mathbb{R}) \rightarrow 2^{\mathbb{R}^m}$  where  $\bar{C}(\mu, \nu) := \{x \in \mathbb{R}^m : f(x, z) \succeq_1 d, x \in X\}$ . Indeed, if  $\nu_n$  converge to  $\nu$  in Kolmogorov-Smirnov sense, then  $\nu_n[d \leq \eta] \rightarrow \nu[d \leq \eta]$  for all  $\eta \in \mathbb{R}$ , and the above proof readily extends.*

**Remark 2.3** (About weak convergence on  $\mathcal{P}(\mathbb{R}^s)$ .) For a different class of random variables, [9] have established a closedness result for a dominance constraint of first order, where convergence on the counterpart space to  $\mathcal{P}(\mathbb{R}^s)$  is given by a suitable discrepancy. Compared with [9], Proposition 2.1 applies to a more focussed family of random variables (even allowing for discontinuities) with a weaker convergence notion on  $\mathcal{P}(\mathbb{R}^s)$ , namely weak convergence of probability measures instead of convergence induced by the discrepancy in [9].

**Remark 2.4** (About weak convergence on  $\mathcal{P}(\mathbb{R})$ .) Proposition 2.1 breaks down when considering variable  $\nu$  and equipping  $\mathcal{P}(\mathbb{R})$  with weak convergence of probability measures. To see this, let  $\eta = 0$  and  $\mu, x, \Phi$  be such that  $1 - \mu[M_0(x)] = 1/2$ . Let  $\nu_n, \nu$  be the discrete probability measures with mass 1 at  $1/n$  and 0, respectively. Then  $\nu_n$  converges weakly to  $\nu$  and we have  $0 = \nu_n[d \leq 0] \leq 1 - \mu[M_0(x)] = 1/2$ , for all  $n$ . On the other hand,  $1 = \nu[d \leq 0] \not\leq 1 - \mu[M_0(x)] = 1/2$ .

Proposition 2.1 in particular implies that  $C(\mu)$  is a closed set for any  $\mu \in \mathcal{P}(\mathbb{R}^s)$ .

**Corollary 2.5** Assume (A1) and (A2). Then  $C(\mu)$  is a closed subset of  $\mathbb{R}^m$  for any  $\mu \in \mathcal{P}(\mathbb{R}^s)$ .

The optimization problem (8) thus is well-posed in the sense that for, e.g., lower semicontinuous  $g$ , bounded  $X$ , and nonempty  $C(\mu)$ , the infimum is finite and is attained.

It is well-known that continuity properties of constraint set mappings, such as the one established in Proposition 2.1, allow for direct conclusions regarding the stability of the related optimization problems. We end this section with such a conclusion.

Consider (8) as a parametric program where the probability distribution  $\mu$  of the random variable  $z(\omega)$  enters as parameter:

$$P(\mu) \quad \min\{g(x) : x \in C(\mu)\}.$$

Studying the stability of stochastic programs with respect to perturbations of the underlying probability distributions is motivated by the incomplete information on these distributions that is often met, and by approximation issues in the context of computations, see [24] for a recent overview on stability analysis in stochastic programming.

**Proposition 2.6** Assume (A1), (A2), that  $X$  is nonempty and compact, and that  $g$  is lower semicontinuous. Let  $\bar{\mu} \in \mathcal{P}(\mathbb{R}^s)$  be such that  $P(\bar{\mu})$  has an optimal solution. Then the optimal value function  $\varphi(\mu) := \inf\{g(x) : x \in C(\mu)\}$  is lower semicontinuous at  $\bar{\mu}$ .

**Proof:** Let  $\mu_n \xrightarrow{w} \bar{\mu}$  and assume without loss of generality that  $C(\mu_n) \neq \emptyset$  for all  $n$ . Otherwise, we would have  $\varphi(\mu_n) = +\infty$  which does not interfere with the validity of  $\liminf_n \varphi(\mu_n) \geq \varphi(\bar{\mu})$ .

Let  $\varepsilon > 0$  be arbitrarily fixed. Then there exist  $x_n \in C(\mu_n)$  such that  $g(x_n) \leq \varphi(\mu_n) + \varepsilon$ . By compactness of  $X$  there exists an accumulation point  $\bar{x}$  of the  $x_n$ . By the closedness of  $C(\cdot)$  (Proposition 2.1), it follows that  $\bar{x} \in C(\bar{\mu})$ . Together with the lower semicontinuity of  $g$  this implies

$$\varphi(\bar{\mu}) \leq g(\bar{x}) \leq \liminf_n g(x_n) \leq \liminf_n \varphi(\mu_n) + \varepsilon.$$

Since  $\varepsilon > 0$  was arbitrary, the proof is complete.  $\square$

### 3 Algorithmic Treatment

In the present section we deal with algorithmic possibilities for (8) in case  $z(\omega)$  and  $d(\omega)$  follow discrete probability distributions with finitely many realizations. We start with establishing an equivalence between (8) and a large-scale, but structured, mixed-integer linear program.

**Proposition 3.1** Let  $z(\omega)$  and  $d(\omega)$  in (8) follow discrete distributions with realizations  $z_l$ ,  $l = 1, \dots, L$ , and  $d_k$ ,  $k = 1, \dots, K$ , as well as probabilities  $\pi_l$ ,  $l = 1, \dots, L$ , and  $p_k$ ,  $k = 1, \dots, K$ , respectively. Let further  $g(x) := g^\top x$  be linear and  $X$  be bounded. Assume (A1) and (A2). Then there exists a constant

M such that (8) is equivalent to the mixed-integer linear program

$$\min \left\{ g^\top x : \begin{array}{ll} c^\top x + q^\top y_{lk} - d_k & \leq M\theta_{lk} \quad \forall l \forall k \\ Tx + Wy_{lk} & = z_l \quad \forall l \forall k \\ \sum_{l=1}^L \pi_l \theta_{lk} & \leq \bar{d}_k \quad \forall k \\ x \in X, y_{lk} \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}, \theta_{lk} \in \{0,1\} & \forall l \forall k \end{array} \right\} \quad (15)$$

where  $\bar{d}_k := 1 - \nu[d \leq d_k]$ ,  $k = 1, \dots, K$ .

**Proof:** By (7), the constraint  $f(x, z) \succeq_1 d$  is equivalent to

$$\nu[d \leq \eta] \leq \mu[f(x, z) \leq \eta] \quad \forall \eta \in \mathbb{R}. \quad (16)$$

We claim that this is equivalent to

$$\nu[d \leq d_k] \leq \mu[f(x, z) \leq d_k] \quad \text{for } k = 1, \dots, K. \quad (17)$$

The necessity of (17) is obvious. For sufficiency assume that the  $d_k$  are arranged in ascending order and consider  $\eta$  with  $d_k \leq \eta < d_{k+1}$ . Then we have

$$\nu[d \leq \eta] = \nu[d \leq d_k] \leq \mu[f(x, z) \leq d_k] \leq \mu[f(x, z) \leq \eta].$$

Here, the first identity is valid since  $d$  is discrete and since there are no mass points in between  $d_k$  and  $d_{k+1}$ . The second relation is just (17), and the last inequality holds due to the monotonicity of the cumulative distribution function. For  $\eta < d_1$  the left-hand side of (16) is zero, so (16) is valid. For  $\eta > d_K$  the validity of (17) for  $k = K$  together with the monotonicity of the cumulative distribution function force the right-hand side of (16) to be one, so (16) is valid. This verifies the asserted equivalence.

Let us now turn our attention to the construction of M. We put M such that

$$M > \sup \{ c^\top x + \Phi(z_l - Tx) - d_k : x \in X, l \in \{1, \dots, L\}, k \in \{1, \dots, K\} \}.$$

It has to be shown that the right-hand side above is finite. To this end, we employ the following growth property of  $\Phi$ , see [2, 6] for instance. Under (A1), (A2) there exist constants  $\alpha > 0, \beta > 0$  such that for all  $t_1, t_2 \in \mathbb{R}^s$

$$|\Phi(t_1) - \Phi(t_2)| \leq \alpha \|t_1 - t_2\| + \beta.$$

Moreover, (A2) implies that  $\Phi(0) = 0$ . This enables the following estimate

$$\begin{aligned} |c^\top x + \Phi(z_l - Tx) - d_k| &\leq |c^\top x| + |\Phi(z_l - Tx) - \Phi(0)| + |d_k| \\ &\leq \|c\| \cdot \|x\| + \alpha \|z_l - Tx\| + \beta + |d_k| \\ &\leq \|c\| \cdot \|x\| + \alpha \|z_l\| + \alpha \|T\| \cdot \|x\| + \beta + |d_k| \end{aligned}$$

Since  $X$  is bounded, this verifies finiteness of the above supremum.

By considering the complementary event on the right, we rewrite (17) as

$$\mu[f(x, z) > d_k] \leq 1 - \nu[d \leq d_k] =: \bar{d}_k \quad \text{for } k = 1, \dots, K. \quad (18)$$

For any  $k \in \{1, \dots, K\}$  we now consider the following sets:

$$S_1 := \{x \in X : \mu[f(x, z) > d_k] \leq \bar{d}_k\}$$

and

$$S_2 := \left\{ x \in X : \begin{array}{l} \exists \theta_l \in \{0,1\} \\ \exists y_l \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}, l = 1, \dots, L, \\ \text{such that:} \\ c^\top x + q^\top y_l - d_k \leq M\theta_l \\ Tx + Wy_l = z_l \\ \sum_{l=1}^L \pi_l \theta_l \leq \bar{d}_k \end{array} \right\}$$

We complete the proof by showing that  $S_1 = S_2$  and begin with the inclusion  $S_1 \subseteq S_2$  :  
Let  $x \in S_1$  and consider  $I := \{l \in \{1, \dots, L\} : c^\top x + \Phi(z_l - Tx) > d_k\}$ . Then  $\sum_{l \in I} \pi_l \leq \bar{d}_k$ , by the definition of  $S_1$ . Put  $\theta_l := 1$  for  $l \in I$ , and  $\theta_l := 0$ , otherwise. This gives

$$\sum_{l=1}^L \pi_l \theta_l = \sum_{l \in I} \pi_l \leq \bar{d}_k.$$

For  $l \notin I$  we have  $c^\top x + \Phi(z_l - Tx) \leq d_k$ . Hence there exists  $y_l \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}$  fulfilling

$$c^\top x + q^\top y_l - d_k \leq 0 = M\theta_l \quad \text{and} \quad Tx + Wy_l = z_l.$$

For  $l \in I$  take  $y_l \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}$  such that  $Tx + Wy_l = z_l$  and  $q^\top y_l = \Phi(z_l - Tx)$ . By the selection of  $M$  we then have

$$c^\top x + q^\top y_l - d_k \leq M = M\theta_l.$$

This implies  $x \in S_2$ .

To show  $S_2 \subseteq S_1$  let  $x \in S_2$  and consider  $I := \{l \in \{1, \dots, L\} : \theta_l = 0\}$ . For each  $l \in I$  then there exists a  $y_l \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}$  such that

$$c^\top x + q^\top y_l - d_k \leq 0 \quad \text{and} \quad Tx + Wy_l = z_l.$$

Hence  $c^\top x + \Phi(z_l - Tx) \leq d_k$  for all  $l \in I$ . Therefore

$$\{l \in \{1, \dots, L\} : c^\top x + \Phi(z_l - Tx) > d_k\} \subseteq \{l \in \{1, \dots, L\} : \theta_l = 1\}.$$

This yields

$$\mu[c^\top x + \Phi(z - Tx) > d_k] \leq \sum_{l \notin I} \pi_l \theta_l = \sum_{l=1}^L \pi_l \theta_l \leq \bar{d}_k.$$

Thus  $x \in S_1$ , and the proof is complete.  $\square$

**Remark 3.2** *As a particular result, the above proof has delivered that, with finite probability spaces, the dominance constrained stochastic program (8) is equivalent to the following optimization problem with finitely many probabilistic (or chance) constraints*

$$\min\{g^\top x : x \in X, \mu[f(x, z) \leq d_k] \geq \nu[d \leq d_k], k = 1, \dots, K\}.$$

*Compared with more traditional chance constrained stochastic programs, an additional difficulty arises, since here  $f(x, z)$  is given by the value function of another extremal problem and has rather poor analytical properties. The fact that, with finite probability spaces, stochastic dominance of first order reduces to a finite number of probabilistic constraints has also been observed in [22] where different classes of random variables are considered and dominance is based on preference of big outcomes over small.*

As a mixed-integer linear program, the optimization problem from Proposition 3.1 clearly can be tackled by general-purpose mixed-integer linear programming software. With growing numbers  $L$  and  $K$  of scenarios of the data and the benchmark distributions, however, it can be expected that this approach will come to its limitations.

This motivates to study decomposition of the model. Having in mind the L-shaped form of the constraint matrix that arises with discrete probability spaces in the traditional stochastic program (5), see [5, 16, 23, 25], similarities and differences come to the fore: The constraints

$$\begin{aligned} c^\top x + q^\top y_{lk} - d_k &\leq M\theta_{lk} \quad \forall l \forall k \\ Tx + Wy_{lk} &= z_l \quad \forall l \forall k \end{aligned}$$

correspond to  $K$  blocks, each of them in L-shaped form. By the latter we mean that, for fixed  $k$ , there are no constraints explicitly interlinking variables  $y_{lk}, \theta_{lk}$  belonging to different  $l \in \{1, \dots, L\}$ . Linkage

is only established by the omnipresent  $x$ -variables. These variables must neither depend on  $l$  nor  $k$ . So they couple the  $K$  blocks above into a single L-shaped block. The constraints

$$\sum_{l=1}^L \pi_l \theta_{lk} \leq \bar{d}_k \quad \forall k \quad (19)$$

provide linkage between variables belonging to different scenarios  $l \in \{1, \dots, L\}$  such that the full model no longer obeys L-shaped structure.

Our basic algorithmic idea now is to generate lower bounds by a suitable relaxation, to generate upper bounds by a tailored feasibility heuristics, and to embed the two into a branch-and-bound scheme in the spirit of global optimization.

#### Lower Bounds:

Relaxation is carried out in a twofold manner: The nonanticipativity of  $x$  gets relaxed by introducing copies  $x_l, l = 1, \dots, L$ . The constraints (19) undergo Lagrangean relaxation. This is formalized as follows. In the objective we put  $x = \sum_{l=1}^L \pi_l x_l$ , and for the constraints (19) we introduce Lagrangean multipliers  $\lambda_k \geq 0, k = 1, \dots, K$ . The Lagrangean function then reads

$$\begin{aligned} \mathcal{L}(x, \theta, \lambda) &= \sum_{l=1}^L \pi_l \cdot g^\top x_l + \sum_{k=1}^K \lambda_k \left( \sum_{l=1}^L \pi_l \theta_{lk} - \bar{d}_k \right) \\ &= \sum_{l=1}^L \pi_l \cdot g^\top x_l + \sum_{l=1}^L \sum_{k=1}^K \lambda_k \cdot (\pi_l \theta_{lk} - \pi_l \bar{d}_k) \\ &= \sum_{l=1}^L \mathcal{L}_l(x_l, \theta_l, \lambda) \end{aligned}$$

where

$$\mathcal{L}_l(x_l, \theta_l, \lambda) := \pi_l \cdot g^\top x_l + \pi_l \sum_{k=1}^K \lambda_k \cdot (\theta_{lk} - \bar{d}_k).$$

This leads to the Lagrangean dual

$$\max\{D(\lambda) : \lambda \in \mathbb{R}_+^K\}$$

where

$$D(\lambda) = \min \left\{ \mathcal{L}(x, \theta, \lambda) : \begin{array}{ll} c^\top x_l + q^\top y_{lk} - d_k & \leq M \theta_{lk} \quad \forall l \forall k \\ T x_l + W y_{lk} & = z_l \quad \forall l \forall k \\ x_l \in X, y_{lk} \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}, \theta_{lk} \in \{0, 1\} & \forall l \forall k \end{array} \right\}$$

The optimization problem behind  $D(\lambda)$  now is separable in  $l$ , and we obtain

$$D(\lambda) = \sum_{l=1}^L \min \left\{ \mathcal{L}_l(x_l, \theta_l, \lambda) : \begin{array}{ll} c^\top x_l + q^\top y_{lk} - d_k & \leq M \theta_{lk} \quad \forall k \\ T x_l + W y_{lk} & = z_l \quad \forall k \\ x_l \in X, y_{lk} \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}, \theta_{lk} \in \{0, 1\} & \forall k \end{array} \right\} \quad (20)$$

The Lagrangean dual is a nonsmooth concave maximization (or convex minimization) problem whose optimal value yields a lower bound to the optimal value of the mixed-integer linear program in Proposition 3.1. For solving the Lagrangean dual, bundle-trust algorithms from nonsmooth convex optimization, such as the conic bundle method [15], can be employed. Per iteration, these methods require the function value  $D(\lambda)$  and one subgradient from  $\partial D(\lambda)$ . Here, the above separability becomes essential, since it

leads to a decomposition of the optimization problem behind  $D(\lambda)$  into subproblems corresponding to the individual scenarios  $z_l$ ,  $l = 1, \dots, L$ .

In principle, the above lower bounding procedure can be improved by applying Lagrangean relaxation not only to (19) but also to the nonanticipativity of  $x$  that can be expressed by the system of identities  $x_1 = x_2 = \dots = x_L$ . This, however, leads to a drastic increase of dimension in the Lagrangean dual, namely from  $K$  to  $K + m \cdot (L - 1)$ . Recall that  $L$  is the number of data scenarios  $z_l$  while  $K$  is the number of benchmark scenarios  $d_k$ . It is reasonable to assume that  $L$ , possibly stemming from past observations, is far bigger than  $K$ , possibly stemming from subjective risk perception. Typically,  $L$  can be in the order of several hundreds or even thousands while  $K$  is around 20 or even less. Compared with Lagrangean relaxation of nonanticipativity, see for instance [7], the above dual bounding scheme thus has the advantage that the Lagrangean dual lives in a space of low dimension.

### Upper Bounds:

An upper bound to the optimal value of (15) is computed by the following heuristics that aims at finding a feasible solution to (15). The input of the heuristics consists of the  $x_l$ -parts  $\tilde{x}_l$  of optimal solutions to the single-scenario problems in (20) for optimal or nearly optimal  $\lambda$ .

### Algorithm 3.3

STEP 1:

*Understand  $\tilde{x}_l$ ,  $l = 1, \dots, L$ , as proposals for  $x$  and pick a “reasonable candidate”  $\bar{x}$ , for instance one arising most frequently, or one with minimal  $\mathcal{L}_l(x_l, \theta_l, \lambda)$ , or average the  $\tilde{x}_l$ ,  $l = 1, \dots, L$ , and round to integers if necessary.*

STEP 2:

*Check whether the following problems are feasible for  $l = 1, \dots, L$ :*

$$\min \left\{ g^\top \bar{x} : \begin{array}{ll} c^\top \bar{x} + q^\top y_{lk} - d_k & \leq M\theta_{lk} \\ T\bar{x} + W y_{lk} & = z_l \\ y_{lk} \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}, \quad \theta_{lk} \in \{0, 1\}, \quad k = 1, \dots, K \end{array} \right\} \quad (21)$$

*As soon as one of them fails to be feasible,  $\bar{x}$  cannot be feasible for (15), and the heuristics stops with assigning the formal upper bound  $+\infty$ . Otherwise, go to Step 3.*

STEP 3:

*Check whether the  $\theta_{lk}$  found in (21) fulfil*

$$\sum_{l=1}^L \pi_l \theta_{lk} \leq \bar{d}_k \quad k = 1, \dots, K.$$

*If yes, then a feasible solution to (15) is found. The heuristics stops with the upper bound  $g^\top \bar{x}$ . Otherwise, go to Step 4.*

STEP 4:

*Solve for each  $l = 1, \dots, L$ :*

$$\min \left\{ \sum_{k=1}^K \theta_{lk} : \begin{array}{ll} c^\top \bar{x} + q^\top y_{lk} - d_k & \leq M\theta_{lk} \\ T\bar{x} + W y_{lk} & = z_l \\ y_{lk} \in \mathbb{Z}_+^{\bar{m}} \times \mathbb{R}_+^{m'}, \quad \theta_{lk} \in \{0, 1\}, \quad k = 1, \dots, K \end{array} \right\}$$

*Go to Step 5.*

STEP 5:

*Repeat the test from Step 3 with the  $\theta_{lk}$  found in Step 4. If the test is positive then the heuristics stops with the upper bound  $g^\top \bar{x}$ . Otherwise, the heuristics stops without a feasible solution to (15) and assigns the formal upper bound  $+\infty$ .*



The purpose of Step 4 is to “push down” the  $\theta_{lk}$  in order to fulfil (19). The implementation is such that Step 4 just continues with the feasible solution found in Step 2. The impact of Step 4 is particularly striking, if in Step 2 the  $\theta_{lk}$  were “poorly” selected such that the test in Step 3 fails, although  $\bar{x}$  is feasible for (15), with different  $\theta_{lk}$  though.

#### Branch-and-Bound:

The bounding procedures developed above are integrated into a branch-and-bound scheme where branching is accomplished by partitioning the set  $X$  with increasing granularity. Linear inequalities are used for this purpose, to maintain the mixed-integer linear description of problems.

This results in the following algorithm. By  $\mathbf{P}$  we denote a list of problems, and  $\varphi_{LB}(P)$  is a lower bound for the optimal value of  $P \in \mathbf{P}$ . Moreover,  $\bar{\varphi}$  denotes the currently best upper bound to the optimal value of (15), and  $X(P)$  is the element in the partition of  $X$  belonging to  $P$ .

#### **Algorithm 3.4**

STEP 1 (INITIALIZATION):

Let  $\mathbf{P} := \{(15)\}$  and  $\bar{\varphi} := +\infty$ .

STEP 2 (TERMINATION):

If  $\mathbf{P} = \emptyset$  then the  $\bar{x}$  that yielded  $\bar{\varphi} = g^\top \bar{x}$  is optimal.

STEP 3 (BOUNDING):

Select and delete a problem  $P$  from  $\mathbf{P}$ . Compute a lower bound  $\varphi_{LB}(P)$  by the bounding procedure developed above, and apply Algorithm 3.3 to find a feasible point  $\bar{x}$  of  $P$ .

STEP 4 (PRUNING):

If  $\varphi_{LB}(P) = +\infty$  (infeasibility of a subproblem in (20)) or  $\varphi_{LB}(P) > \bar{\varphi}$  (inferiority of  $P$ ), then go to Step 2.

If  $\varphi_{LB}(P) = g^\top \bar{x}$  (optimality for  $P$ ), then check whether  $g^\top \bar{x} < \bar{\varphi}$ . If yes, then  $\bar{\varphi} := g^\top \bar{x}$ . Go to Step 2.

If  $g^\top \bar{x} < \bar{\varphi}$ , then  $\bar{\varphi} := g^\top \bar{x}$ .

STEP 5 (BRANCHING):

Create two new subproblems by partitioning the set  $X(P)$ . Add these subproblems to  $\mathbf{P}$  and go to Step 2.

Generally speaking, the branching in Step 5 is accomplished by applying linear inequalities, to maintain representation of subproblems as mixed-integer linear programs. In practice, however, these inequalities usually correspond to ranges of components of variables. For continuous variables, tolerances are used to avoid endless branching with finer and finer granularity.

## **4 Computations**

In the following we report computational results for Algorithm 3.4 applied to test instances from power planning. The first group of instances refers to the optimal management of a dispersed generation (DG) system run by a power utility in Germany, see [14] for a detailed model description. The instances of the second group are inspired by an early stochastic program from the literature, the investment planning problem for electricity generation of [18].

### **4.1 Dispersed Generation System**

The system consists of five engine-based cogeneration (CG) stations, producing power and heat simultaneously, twelve wind turbines and one hydroelectric power plant. The CG stations include eight gas boilers, nine gas motors and one gas turbine, and each is equipped with a thermal storage and a cooling device. While the heat is distributed locally, the electricity is fed into the global distribution network. The cost minimal operation of this system with respect to relevant technical constraints and fulfilment

of heat and power demand can be formulated as a mixed-integer linear program, with on-off decisions for the generation units as source of integrality. With a planning horizon of 24 hours, divided into quarter-hourly subintervals, this (still deterministic) model has about 17500 variables (9000 Boolean and 8500 continuous) and 22000 constraints.

The optimization problem is influenced by stochasticity on the production side, where the infeed from renewable resources is not known with certainty, as well as on the consumers side, where demand of electrical and thermal energy are uncertain. The problem turns into a random mixed-integer linear problem, a specification of (1).

Assuming that the uncertainty prone data is known for the first four hours of the planning horizon leads to a two-stage stochastic program with the decisions belonging to these first four hours as first stage. For a more detailed description of the arising stochastic program and results on purely expectation-based and mean-risk specifications of (6) see [14, 26].

To derive a benchmark profile  $d$  in (8) we first consider  $f(\hat{x}, \omega)$  where  $\hat{x}$  denotes an optimal solution to the expectation model (5). With heuristically selected benchmark values, the  $f(\hat{x}, \omega)$  then are clustered around these values, and the probability of each benchmark value arises as the sum of the probabilities of the members in its cluster. Further problem instances were derived by fixing the probabilities and increasing the values of  $d$  successively.

A meaningful objective function  $g$  is to count the number of start-ups over all units and time steps in the first stage. This number serves as a measure for the abrasion of the DG units. Then the dominance constrained model minimizes abrasion of units over all generation policies incurring costs that, in a stochastic sense, do not exceed the given benchmark profile.

We report results for instances with  $K = 4$  benchmark scenarios and  $L = 10$  up to 50 scenarios for heat and power demand. The deterministic equivalents according to Proposition 3.1 then finally are truly large-scale, as seen in Table 1, and can hardly be handled with mixed-integer solvers like Cplex ([8]).

Number of	10 scenarios	20 scenarios	30 scenarios	50 scenarios
Boolean variables	299159	596799	894439	1489719
continuous variables	283013	564613	846213	1409413
constraints	742648	1481568	2220488	3698328

Table 1: Dimensions of mixed-integer linear programming equivalents

In Tables 2-5 computations for these equivalents with Cplex are compared to computations made with the implementation ddsip.vSD of Algorithm 3.4 derived in Section 3. Problems were solved on a Linux-PC with a 3.2GHz pentium processor and 2GB ram. As stopping criterion we used a timelimit of eight hours.

Number of scenarios	Instance	Benchmarks		Time (sec.)	Cplex		ddsip.vSD	
		Probability	Benchmark Value		Upper Bound	Lower Bound	Upper Bound	Lower Bound
10	1	0.12	2895000	1348.95	–	29	29	18
		0.21	4851000	15325.75	29	29	29	22
		0.52	7789000	28800.00	29	29	29	23
		0.15	10728000					
	2	0.12	2900000	273.78	–	27	28	14
		0.21	4860000	418.90	28	28	28	14
		0.52	7800000	28800.00	28	28	28	22
		0.15	10740000					
	3	0.12	3000000	192.45	–	21	21	12
		0.21	5000000	428.61	21	21	21	12
		0.52	8000000	28800.00	21	21	21	16
		0.15	11000000					
	4	0.12	3500000	227.44	–	11	13	10
		0.21	5500000	2593.35	18	12	13	10
		0.52	8500000	28800.00	13	13	13	11
		0.15	11500000					
	5	0.12	4000000	225.91	–	7	8	8
		0.21	6000000	3304.02	8	8	8	8
		0.52	9000000					
		0.15	12000000					

Table 2: Results for instances with 10 data scenarios and 4 benchmark scenarios

In all tables, the benchmark costs increase successively from instance 1 to instance 5. This means the dominance constraints get easier to fulfil. As one would expect, this affects the needed numbers of start-ups positively. They decrease with increasing reference values, which is reported in the column 'Upper

Number of scenarios	Instance	Benchmarks		Time (sec.)	Cplex		ddsip.vSD	
		Probability	Benchmark Value		Upper Bound	Lower Bound	Upper Bound	Lower Bound
20	1	0.105	2895000	3756.25	–	29	29	12
		0.1	4851000	8951.05	30	29	29	19
		0.69	7789000	28800.00	29	29	29	22
		0.105	10728000					
	2	0.105	2900000	1268.50	–	27	28	13
		0.1	4860000	1744.75	28	28	28	14
		0.69	7800000	28800.00	28	28	28	18
		0.105	10740000					
	3	0.105	3000000	357.99	–	21	21	10
		0.1	5000000	1682.93	21	21	21	11
		0.69	8000000	28800.00	21	21	21	13
		0.105	11000000					
	4	0.105	3500000	1036.50	–	11	14	9
		0.1	5500000	11236.31	–	12 out of mem.	14	10
		0.69	8500000	28800.00	–	–	14	10
		0.105	11500000					
	5	0.105	4000000	4574.40	–	8	8	8
		0.1	6000000	5599.99	9	8	8	8
		0.69	9000000	7840.09	9	8 out of mem.	8	8
		0.105	12000000					

Table 3: Results for instances with 20 data scenarios and 4 benchmark scenarios

Bound’, where the objective value of the current best solution is displayed. The corresponding best lower bound can be found in the column ‘Lower Bound’.

In every table, we show the status of the optimization for different points in time. Usually the first two points are the times, where either the decomposition method or Cplex find the first feasible solution. Also for the timelimit of eight hours the objective values and the best bounds are given for each solver, unless optimality was proven earlier.

For test instances with 20 or 30 scenarios Cplex sometimes stops before reaching a first feasible solution, because the available memory is exceeded (marked by ‘out of mem.’). In these cases, only the lower bounds already found before the memory error occurred are displayed.

With 50 data scenarios the deterministic equivalents get so large, that the available memory is not sufficient to build up the model (lp-) file used by Cplex, preventing optimization with Cplex for these instances. That’s why the last table only reports best values and lower bounds calculated with the decomposition method ddsip.vSD.

Number of scenarios	Instance	Benchmarks		Time (sec.)	Cplex		ddsip.vSD	
		Probability	Benchmark Value		Upper Bound	Lower Bound	Upper Bound	Lower Bound
30	1	0.085	2895000	2074.96	–	28	29	12
		0.14	4851000	3255.99	–	29 out of mem.	29	15
		0.635	7789000	28800.00	–	–	29	21
		0.14	10728000					
	2	0.085	2900000	1291.00	–	26	28	13
		0.14	4860000	3372.24	–	27 out of mem.	28	13
		0.635	7800000	28800.00	–	–	28	17
		0.14	10740000					
	3	0.085	3000000	569.27	–	17	23	10
		0.14	5000000	3681.15	–	18 out of mem.	23	10
		0.635	8000000	28800.00	–	–	23	12
		0.14	11000000					
	4	0.085	3500000	874.84	–	10	14	8
		0.14	5500000	3095.02	–	11 out of mem.	14	8
		0.635	8500000	28800.00	–	–	14	9
		0.14	11500000					
	5	0.085	4000000	6449.12	–	7	8	8
		0.14	6000000	8504.88	–	8 out of mem.	8	8
		0.635	9000000					
		0.14	12000000					

Table 4: Results for instances with 30 data scenarios and 4 benchmark scenarios

Our computations show, that for all instances the decomposition method reaches the first feasible solution faster than Cplex does. In most cases this is already an optimal solution, with our method having difficulties to prove this, however. As long as not running out of memory, Cplex provides the preferable lower bounds. For the DG-instances the moderate quality of the lower bounds of the decomposition method seems to result from the (quite drastic) relaxation of nonanticipativity. On the other hand, this relaxation enables decomposition and, thus, the handling of large-scale instances where general-purpose

Number of scenarios	Instance	Benchmarks		Time (sec.)	Cplex		ddsip.vSD	
		Probability	Benchmark Value		Upper Bound	Lower Bound	Upper Bound	Lower Bound
50	1	0.09	2895000	21210.30	–	–	29	20
		0.135	4851000	28800.00	–	–	29	20
		0.67	7789000					
		0.105	10728000					
	2	0.09	2900000	1593.48	–	–	28	13
		0.135	4860000	28800.00	–	–	28	17
		0.67	7800000					
		0.105	10740000					
	3	0.09	3000000	1716.13	–	–	23	10
		0.135	5000000	28800.00	–	–	23	12
		0.67	8000000					
		0.105	11000000					
	4	0.09	3500000	7917.97	–	–	15	8
		0.135	5500000	28800.00	–	–	15	9
		0.67	8500000					
		0.105	11500000					
	5	0.09	4000000	21071.54	–	–	8	8
		0.135	6000000					
		0.67	9000000					
		0.105	12000000					

Table 5: Results for instances with 50 data scenarios and 4 benchmark scenarios

MILP solvers run out of memory.

In the computations dealing with 30 and 50 scenarios, the superiority of the decomposition method over general-purpose solvers gets evident. For 30 scenarios Cplex can’t provide any feasible solution, for 50 scenarios even no lower bound, while ddsip.vSD is able to solve the problems - not always to the optimum, but with good solutions.

## 4.2 Investment Planning

The investment planning problems for electricity generation that form our second group of test instances are inspired by [18]. We consider two-stage versions of the multi-stage model there and add integrality requirements to the first stage. This leads to a two-stage mixed-integer linear stochastic program where, in the first stage, decisions on capacity expansions for different generation technologies under budget constraints and supply guarantee are made. We assume that these decisions reflect indivisibilities (generation units) and hence are integer-valued. The second stage concerns the minimization of production costs for electricity under the constraints that electricity demand is met and the available capacity is not exceeded.

The electricity demand is captured by a load duration curve assigning to each duration  $\tau \in \mathbb{R}_+$  the minimum load to be covered over time spans adding up to  $\tau$ . This is where uncertainty enters, since in praxis load durations are typically available only stochastically. The model uses step function approximations for load duration curves. So each data scenario is represented by a (finite) step function.

The aim of the optimization is cost minimization where costs are incurred by the expansion decisions of the first stage and the production levels of the second stage. Together with the random load durations this leads to a random optimization problem which is a specification of (1).

The benchmarks were constructed in a similar way as in Subsection 4.1. With first-stage decisions  $x$  fixed to “reasonable” values, the  $f(x, \omega)$  were clustered around heuristically selected benchmark values, whose probabilities were obtained as probabilities of the cluster sets.

As objective function  $g$  we considered the capacity expansion of one of the different technologies, possibly one least desired for environmental reasons. The dominance constrained stochastic program then minimizes expansion of this capacity over all expansion policies whose costs do not exceed the benchmark profile in terms of first-order stochastic dominance.

We report results for instances with  $K = 3$  up to 20 benchmark scenarios and  $L = 20$  up to 500 scenarios for load duration. Deterministic equivalents according to Proposition 3.1 again become pretty large-scale. Table 6 shows dimensions for  $K = 20$  and the different  $L$ .

Table 7 summarizes our computations for the investment planning instances. Again, a Linux-PC with a 3.2GHz pentium processor and 2GB ram was used. The time limit was set to one hour.

The first column indicates three principal problem instances marked by their optimal values. (Let us remark that all test instances were constructed in such a way that their optimal values were known in advance.) The next two columns list the numbers  $K$  of benchmark and  $L$  of data scenarios. The remaining

Number of	20 scenarios	50 scenarios	100 scenarios	500 scenarios
Boolean and integer variables	404	1004	2004	10004
continuous variables	38400	96000	192000	960000
constraints	11622	29022	58022	290022

Table 6: Dimensions of mixed-integer linear programming equivalents

columns list lower and upper bounds obtained when applying Cplex [8] and our implementation ddsip.vSD of Algorithm 3.4. Time entries deviating from the limit of 1h indicate that the instance was solved to optimality within this span.

Optimal Value	Number of Benchmark Scenarios	Number of Data Scenarios	Cplex			ddsip.vSD		
			Upper Bound	Lower Bound	Time	Upper Bound	Lower Bound	Time
0	3	20	0	0	17s	0	0	69s
		50	0	0	2712s	0	0	138s
		100	—	0	1h	0	0	718s
		500	—	0	1h	0	0	2162s
	10	20	0	0	3197s	0	0	70s
		50	—	0	1h	0	0	588s
		100	—	0	1h	0	0	2327s
		500	—	0	1h	8	0	1h
	20	20	—	0	1h	0	0	368s
		50	—	0	1h	0	0	2395s
		100	—	0	1h	23	0	1h
		500	—	0	1h	166	0	1h
1	3	20	1	1	15.9s	1	1	659s
		50	—	0	1h	1	1	1244s
		100	—	0	1h	2	1	1h
		500	—	0	1h	3	1	1h
	10	20	—	0.771	1h	1	1	1116s
		50	—	0	1h	4	1	1h
		100	—	0	1h	2	1	1h
		500	—	0	1h	8	0	1h
	20	20	—	0	1h	1	1	3039s
		50	—	0	1h	12	1	1h
		100	—	0	1h	2	0	1h
		500	—	0	1h	170	0	1h
100	3	20	100	100	11.31s	101	72	1h
		50	—	76.6	1h	104	38	1h
		100	—	27	1h	100	33	1h
		500	—	0	1h	111	16	1h
	10	20	—	99.5	1h	101	85	1h
		50	—	40	1h	102	56	1h
		100	—	27	1h	101	44	1h
		500	—	0	1h	102	44	1h
	20	20	—	72	1h	103	92	1h
		50	—	40	1h	207	80	1h
		100	—	27	1h	160	67	1h
		500	—	0	1h	184	54	1h

Table 7: Results for investment planning instances

It becomes evident that, at the investment planning instances, Algorithm 3.4 is superior to applying a general-purpose solver like Cplex. This refers both to upper, and in contrast with the results in Subsection 4.1, also lower bounds. Although we have experimented with various time limits and parameter settings in Cplex, such as “emphasize integer feasibility”, we were unable to improve the Cplex results for upper bounds. The instance 0/3/100 (optimal-value/ $K/L$ ), for example, was solved to optimality by Cplex after more than three hours only. For the instance 0/10/100, as another example, Cplex did not find a feasible solution even after four days of computing time.

**Acknowledgement.** We wish to thank Christoph Helmberg (Technical University of Chemnitz) for giving us access to the implementation of his spectral bundle method. Moreover, we are grateful to Uwe Gotzes (University of Duisburg-Essen) for fruitful discussions and his support in running computational tests. This paper was written while the third author was visiting the Centro de Modelamiento Matematico, Universidad de Chile, Santiago. Partial funding for this research was provided by the German Federal Ministry of Education and Research (BMBF) under grant 03-SCNIVG. We thank these institutions for their support.

## References

- [1] Ahmed, S.: Convexity and decomposition of mean-risk stochastic programs, *Mathematical Programming* 106 (2006), 433–446.
- [2] Bank, B.; Mandel, R.: *Parametric Integer Optimization*, Akademie-Verlag, Berlin, 1988.
- [3] Billingsley, P.: *Convergence of Probability Measures*, Wiley, New York, 1968.
- [4] Billingsley, P.: *Probability and Measure*, Wiley, New York, 1986.
- [5] Birge, J.R.; Louveaux, F.: *Introduction to Stochastic Programming*, Springer, New York, 1997.
- [6] Blair, C.E.; Jeroslow, R.G.: The value function of a mixed integer program: I. *Discrete Mathematics* 19 (1977), 121–138.
- [7] Carøe, C.C.; Schultz, R.: Dual decomposition in stochastic integer programming. *Operations Research Letters* 24 (1999), 37–45.
- [8] CPLEX Callable Library 9.1.3, ILOG (2005).
- [9] Dentcheva, D.; Henrion, R.; Ruszczyński, A.: Stability and sensitivity of optimization problems with first order stochastic dominance constraints, *SIAM Journal on Optimization* (submitted).
- [10] Dentcheva, D.; Ruszczyński, A.: Stochastic optimization with dominance constraints, *SIAM Journal on Optimization* 14 (2003), 548–566.
- [11] Dentcheva, D.; Ruszczyński, A.: Optimality and duality theory for stochastic optimization with nonlinear dominance constraints, *Mathematical Programming* 99 (2004), 329–350.
- [12] Eichhorn, A.; Römisch, W.: Polyhedral risk measures in stochastic programming, *SIAM Journal on Optimization* 16 (2005), 69–95.
- [13] Fishburn, P.C.: *Utility Theory for Decision Making*, Wiley, New York, 1970.
- [14] Handschin, E.; Neise, F.; Neumann, H.; Schultz, R.: Optimal operation of dispersed generation under uncertainty using mathematical programming, *International Journal of Electrical Power & Energy Systems* 28 (2006), 618–626.
- [15] Helmberg, C.; Kiwiel, K.C.: A spectral bundle method with bounds, *Mathematical Programming* 93 (2002), 173–194.
- [16] Kall, P.; Wallace, S.W.: *Stochastic Programming*, Wiley, Chichester, 1994.
- [17] Kristoffersen, T.K.: Deviation measures in two-stage stochastic linear programming, *Mathematical Methods of Operations Research* 62 (2006), 255–274.
- [18] Louveaux, F.V.; Smeers, Y.: Optimal investments for electricity generation: A stochastic model and a test problem, in: *Numerical Techniques for Stochastic Optimization* (Yu. Ermoliev, R.J-B Wets, eds.), Springer, Berlin 1988, 445–453.
- [19] Märkert, A.; Schultz, R.: On deviation measures in stochastic integer programming, *Operations Research Letters* 33 (2005), 441–449.
- [20] Mann, H.B.; Whitney, D.R.: On a test of whether one of two random variables is stochastically larger than the other, *Annals of Mathematical Statistics* 18 (1947), 50–60.
- [21] Müller, A.; Stoyan, D.: *Comparison Methods for Stochastic Models and Risks*, Wiley, Chichester, 2002.
- [22] Noyan, N.; Rudolf, G.; Ruszczyński, A.: Relaxations of linear programming problems with first order stochastic dominance constraints, *Operations Research Letters* 34 (2006), 653–659.
- [23] Prékopa, A.: *Stochastic Programming*, Kluwer, Dordrecht, 1995.

- [24] Römisch, W.: Stability of stochastic programming problems, in [25], 483–554.
- [25] Ruszczyński, A.; Shapiro, A. (eds.): Handbooks in Operations Research and Management Science, 10: Stochastic Programming, Elsevier, Amsterdam, 2003.
- [26] Schultz, R.; Neise, F.: Algorithms for mean-risk stochastic integer programs in energy, Preprint 615-2005, Department of Mathematics, University of Duisburg-Essen, Investigación Operacional (to appear).
- [27] Schultz, R.; Tiedemann, S.: Risk Aversion via Excess Probabilities in Stochastic Programs with Mixed-Integer Recourse. *SIAM Journal on Optimization* 14 (2003), 115–138.
- [28] Schultz, R.; Tiedemann, S.: Conditional value-at-risk in stochastic programs with mixed-integer recourse, *Mathematical Programming* 105 (2006), 365–386.